2007 IEEE Workshop on Nonlinear Circuit Networks
NCN'07, Tokushima, Japan, Dec 7--8, 2007

Clustering and Feature Extraction of SOM with False-Neighbor Degree

Haruna Matsushita and Yoshifumi Nishio
Department of Electrical and Electronic Engineering,
Tokushima University
Email: haruna@ee.tokushima-u.ac.jp, nishio@ee.tokushima-u.ac.jp

***Abstract*— In the real world, it is not always true that the nextdoor house is close to my house, in other words, "neighbors" are not always "true neighbors". In this study, we propose a new Self-Organizing Map (SOM) algorithm, SOM with False-Neighbor degree between neurons (called FN-SOM). The behavior of FN-SOM is investigated with learning for various input data. We confirm that FN-SOM can obtain the more effective map reflecting the distribution state of input data than the conventional SOM and Growing Grid.**

I. Introduction

The Self-Organizing Map (SOM) is an unsupervised neural network [1] and has attracted attention for its clustering properties. In the learning algorithm of SOM, a winner, which is a neuron closest to the input data, and its neighboring neurons are updated, regardless of the distance between the input data and the neighboring neurons. For this reason, if we apply SOM to clustering of the input data which includes some clusters located at distant location, there are some inactive neurons between clusters where without the input data.

Then, what are the "neighbors"? In the real world, it is not always true that the next-door house is close to my house. In other words, "neighbors" are not always "true neighbors". In addition, the relationship between neighborhoods is not fixed, but keeps changing with time. It is important to change the neighborhood relationship flexibly according to the situation.

On the other side, the synaptic strength is not constant in the brain. So far, the Growing Grid network was proposed in 1985 [2]. Growing Grid increases the neighborhood distance between neurons by increasing the number of neurons. However, there are few researches changing the synaptic strength as far as we know even though there are algorithms which increase the number of neurons or consider rival neurons [3].

In our past study, we proposed the algorithm which changes the neighborhood distance between neurons [4]. However, the algorithm used the rank order of the distances between the input data and weight vectors in addition to changing the neighborhood distance. Thus the algorithm did not work well if the positions of all the weight vectors of the neurons were not taken into consideration. Moreover, the algorithm needs a lot of calculation amount because we have to calculate the rank order at every updating of the weight vector.

In this study, we propose a new SOM algorithm, SOM with False-Neighbor degree between neurons (called FN-SOM). False-neighbor degrees are allocated between adjacent rows and adjacent columns of FN-SOM. We find the neuron q which has never become the winner, and the neurons, which is the most distant from q in a set of direct topological neighbors of q, are said to be "false neighbors" of q. The initial values of all of the false-neighbor degrees are set to zero, however, they are increased with learning, and the false-neighbor degrees act as a burden of the distance between map nodes when the weight vectors of neurons are updated. FN-SOM changes the neighborhood relationship more flexibly according to the situation and the shape of data.

We explain the learning algorithm of FN-SOM in detail in Section II. The learning behaviors of FN-SOM for 2-dimensional input data and 3-dimensional data, which have some clustering problem, are investigated in Section III. In addition, we apply FN-SOM to a real world data set, Iris data. Learning performance is evaluated both visually and quantitatively using three measurements. Furthermore, the results are compared with those obtained by the conventional SOM and Growing Grid. We can confirm that there are few inactive neurons using FN-SOM, and FN-SOM can obtain the most effective map reflecting the distribution state of input data in the three algorithms.

II. SOM with False-Neighbor Degree (FN-SOM)

We explain learning algorithm of SOM with False-Neighbor Degree (FN-SOM). FN-SOM consist of $n \times m$ neurons located at 2-dimensional rectangular grid. Each neuron i has a d-dimensional weight vector $\boldsymbol{w}_i = (w_{i1}, w_{i2}, \cdots, w_{id})$ $(i = 1, 2, \cdots, nm)$. The range of the elements of d-dimensional input data $\boldsymbol{x}_j = (x_{j1}, x_{j2}, \cdots, x_{jd})$ $(j = 1, 2, \cdots, N)$ are assumed to be from 0 to 1. False-neighbor degrees of rows R_r $(1 \leq r \leq n-1)$ are allocated between adjacent rows of FN-SOM with the size of $n \times m$ grid. Likewise, false-neighbor degrees of columns C_k $(1 \leq k \leq m-1)$ are allocated between adjacent columns of FN-SOM. The initial values of the false-neighbor degrees are set to zero.

Learning Step

(FN-SOM1) An input vector $\boldsymbol{x}_j$ is inputted to all the neurons at the same time in parallel.

(FN-SOM2) Distances between $\boldsymbol{x}_j$ and all the weight vectors are calculated, The winner, denoted by c, is the neuron with the weight vector closest to the input vector $\boldsymbol{x}_j$:

$$c = \arg\min_i \{\|\boldsymbol{w}_i - \boldsymbol{x}_j\|\}, \tag{1}$$

where $\|\cdot\|$ is the distance measure, Euclidean distance.

(FN-SOM3) Increment of the winning frequency of winner c by ${\gamma_c}^{\text{new}} = {\gamma_c}^{\text{old}} + 1$.

(FN-SOM4) The neighboring distances between the winner c and the other neurons are calculated. For instance, for two neurons s_1, which is located at r_1-th row and k_1-th column, and s_2, which is located at r_2-th row and k_2-th column, the neighboring distance is defined as the following measure:

$$d_f(s_1, s_2) = \left(|r_1 - r_2| + \sum_{r=r_1}^{r_2-1} R_r \right)^2 + \left(|k_1 - k_2| + \sum_{k=k_1}^{k_2-1} C_k \right)^2, \quad (2)$$

where $r_1 < r_2$, $k_1 < k_2$, namely, $\sum_{r=r_1}^{r_2-1} R_r$ means the sum of the false-neighbor degrees between the rows r_1 and r_2, and $\sum_{k=k_1}^{k_2-1} C_k$ means the sum of the false-neighbor degrees between the column k_1 and k_2.

(FN-SOM5) The weight vectors of the neurons are updated:

$$\boldsymbol{w}_i(t+1) = \boldsymbol{w}_i(t) + h_{Fc,i}(t)(\boldsymbol{x}_j - \boldsymbol{w}_i(t)), \quad (3)$$

where $h_{Fc,i}(t)$ is the neighborhood function of FN-SOM:

$$h_{Fc,i}(t) = \alpha(t) \exp\left(-\frac{d_f(c,i)}{2\sigma^2(t)} \right). \quad (4)$$

Both $\alpha(t)$ and $\sigma(t)$ decrease with time.

(FN-SOM6) If $\sum_{i=1}^{nm} \gamma_i \geq \lambda$ is satisfied, we find the false-neighbors and increase the false-neighboring degree, according to steps from (FN-SOM7) to (FN-SOM10). If not, we perform step (FN-SOM11). In other words, we consider the false-neighbors every time when the learning steps are performed for λ input data.

Considering False-Neighbors

(FN-SOM7) We find a set of neurons S which have never become the winner: $S = \{i \mid \gamma_i = 0\}$. Some false-neighbors are found in one considering false-neighbor step. If the neuron, which have never become the winner, does not exist, namely $|S| = 0$, we return to (FN-SOM1) without considering the false-neighbors. It leads to reduce the amount of computation time, and FN-SOM can change the neighborhood relationship more flexibly.

(FN-SOM8) A false-neighbor f_q of each neuron q in S $(q = 1, 2, \cdots, |S|)$ is chosen from the set of direct topological neighbors of q denoted as N_{q1}. f_q is the neuron whose weight vector is most distant from q:

$$f_q = \arg\max_i \{\|\boldsymbol{w}_i - \boldsymbol{w}_q\|\}, \quad q \in S, \ i \in N_{q1}. \quad (5)$$

(FN-SOM9) A false-neighbor degree between each q and its false-neighbor f_q, R_r or C_k, is increased. If q and f_q are in the r-th row and in the k-th and $(k+1)$-th column (as Fig. 1(a)), the false-neighbor degree C_k between columns k and $k+1$ is increased according to

$${C_k}^{\text{new}} = {C_k}^{\text{old}} + \frac{n+m}{2nm}. \quad (6)$$

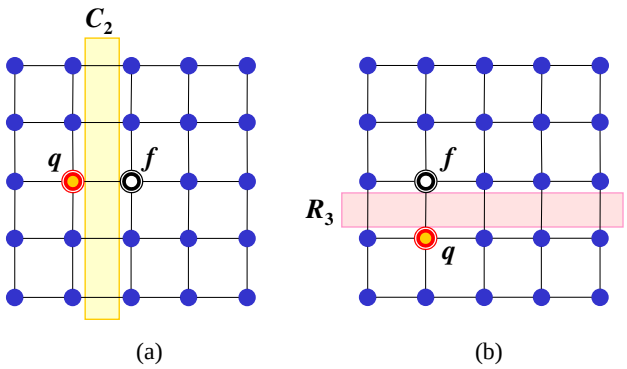


Fig. 1. Increment of the false-neighbor degree. (a) q and its false-neighbor f_q are in the 3rd row and in the 2nd and 3rd column, respectively. Then, the false-neighbor degree C_2 between columns 2 and 3 is increased by Eq. (6). (b) q and f_q are in the 2nd column and in the 4th and 3rd row, respectively. The false-neighbor degree R_3 between rows 3 and 4 is increased by Eq. (7).

In the same way, if q and f_q are in the k-th column and in the $(r+1)$-th and r-th row (as Fig. 1(b)), the false-neighbor degree R_r between rows r and $r+1$ is also increased according to

$${R_r}^{\text{new}} = {R_r}^{\text{old}} + \frac{n+m}{2nm}. \quad (7)$$

These amounts of increasing the false-neighbor degree are derived by the number of neurons numerically and are fixed.

(FN-SOM10) The winning frequency of all the neurons are reset to zero: $\gamma_i = 0$.

(FN-SOM11) The steps from (FN-SOM1) to (FN-SOM10) are repeated for all the input data.

III. Experimental Results

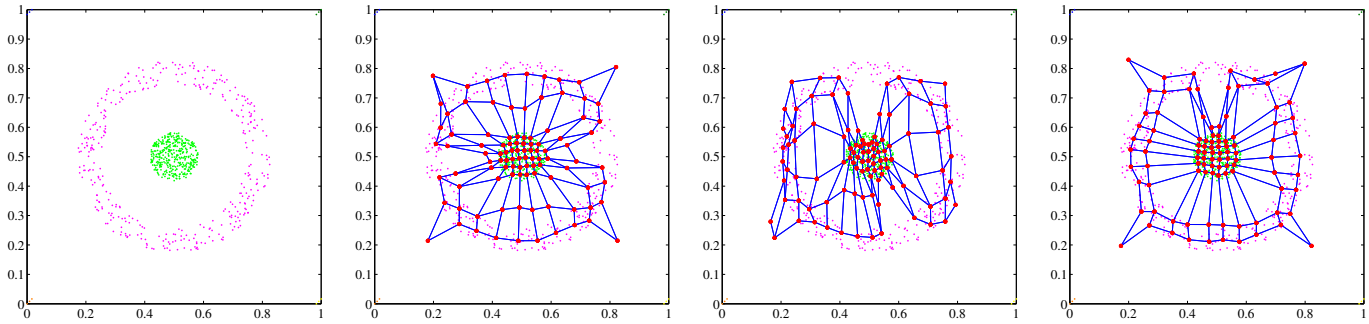
We apply FN-SOM to various input data and compare FN-SOM with the conventional SOM and Growing Grid.

A. For 2-dimensional data

First, we consider 2-dimensional input data as shown in Fig. 2(a). The input data is Target data set, which has a clustering problem of outliers [5]. Total number of the input data N is 770, and the input data has six clusters which include 4 outliers. All the input data are sorted at random.

Both the conventional SOM and FN-SOM has $nm = 100$ neurons (10×10). Growing Grid starts learning with a 2×2 neurons, and new rows and columns are inserted as long as the number of neurons is less than $nm_{\max} = 100$. We repeat the learning 20 times for all input data, namely $t_{\max} = 15400$. $\lambda = 3000$ for FN-SOM are chosen. We use the same decreasing equations of α and σ (in Eq. (4)) to SOM and FN-SOM for the comparison and the confirmation of the false-neighbor degree effect. The input data are normalized and are sorted at random.

The learning results of the conventional SOM and Growing Grid are shown in Figs. 2(b) and (c), respectively. We can see that there are some inactive neurons between clusters. The other side, the result of FN-SOM is shown in Fig. 2(d). We can see from this figure that there are just a few inactive neurons



(a) (b) (c) (d)

Fig. 2. Learning results of three algorithms for Target data. (a) Input data. (b) Conventional SOM. (c) Growing Grid. (d) FN-SOM.

between clusters, and FN-SOM can obtain the more effective map reflecting the distribution state of input data than SOM and Growing Grid.

Furthermore, in order to evaluate the learning performance of FN-SOM in comparison with the conventional SOM and Growing Grid, we use the following three measurements to evaluate the training performance of the three algorithms.

Quantization Error *Qe* measures the average distance between each input vector and its winner [1]. The small value Qe is more desirable.

Topographic Error *Te* describes how well the SOM preserves the topology of the studied data set [6]. The small value Te is more desirable. Unlike the quantization error, it considers the structure of the map. For a strangely twisted map, the topographic error is big even if the quantization error is small.

Neuron Utilization *U* measures the percentage of neurons that are the winner of one or more input vector in the map [3]. Thus, U nearer 1.0 is more desirable.

The calculated three measurements are shown in Table. I. The quantization error Qe of FN-SOM is the smallest value in the three algorithms, and by using FN-SOM, the quantization error Qe has improved 7.7% from using the conventional SOM. This is because the result of FN-SOM has few inactive neurons, therefore, the more neurons can self-organize the input data. This is confirmed by the neuron utilization U. The neuron utilization U of FN-SOM is the largest value in the three algorithms. It means that 90% neurons of FN-SOM are the winner of one or more input data, namely, there are few inactive neurons. On the other hand, the topographic error Te of FN-SOM is the smallest value although Qe and U are the best values. It means that FN-SOM self-organizes most effectively with maintenance of top quality topology.

B. For 3-dimensional data

Next, we apply three algorithm to 3-dimensional input data, Hepta data set [5], as shown in Fig. 3(a). The input data has a clustering problem of different variances. Total number of the input data N is 212, and the input data has seven clusters. All the input data are sorted at random.

TABLE I

QUANTIZATION ERROR Qe, TOPOGRAPHIC ERROR Te AND NEURON UTILIZATION U FOR TARGET DATA.

	SOM	Growing Grid	FN-SOM
Qe	0.0207	0.0237	0.0191
Te	0.0740	0.2455	0.0442
U	0.8100	0.8137	0.9100

TABLE II

QUANTIZATION ERROR Qe, TOPOGRAPHIC ERROR Te AND NEURON UTILIZATION U FOR HEPTA DATA.

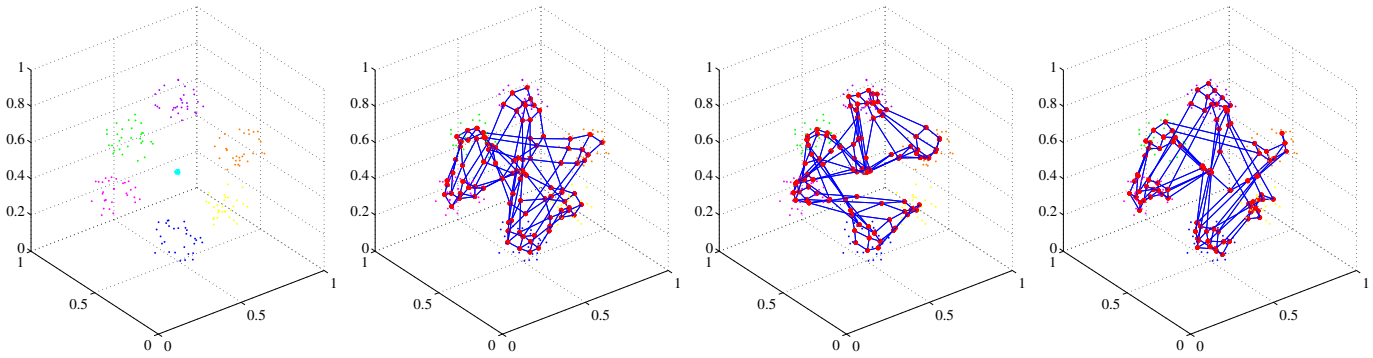
	SOM	Growing Grid	FN-SOM
Qe	0.0360	0.0409	0.0306
Te	0.1698	0.1509	0.1462
U	0.6500	0.6863	0.8000

We repeat the learning 70 times for all input data, namely $t_{\max} = 14840$. The input data are normalized and are sorted at random. The learning conditions are the same used in Fig. 2.

The learning results of three algorithms are shown in Figs. 3(b), (c) and (d), respectively. We can see that FN-SOM has the fewest inactive neurons between clusters. Three map quality measures are shown in Table. II. Results of FN-SOM are the best values in all map quality measures. The quantization error Qe of FN-SOM is the smallest value and has improved 15.0% from using the conventional SOM. Nonetheless, the topographic error Te of FN-SOM is also the smallest value, and it has improved 13.9% from using the conventional SOM and 3.1% from using Growing Grid. The neuron utilization U of FN-SOM is also the best value in three SOMs, and it has improved 23.1% from using the conventional SOM and 16.6% from using Growing Grid. From these results, we can say that FN-SOM is the most effective.

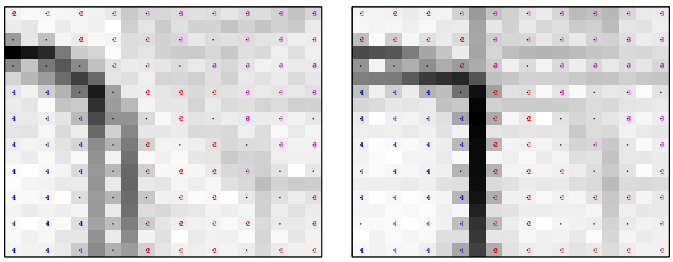
C. For Iris data

Furthermore, we apply FN-SOM to the real world clustering problem. We use the Iris plant data as real data [7]. This data is one of the best known databased to be found in pattern recognition literatures. The data set contains three clusters of 50 instances respectively, where each class refers to a type of iris plant. The number of attributes is four as the sepal length,



(a) (b) (c) (d)

Fig. 3. Learning results of three algorithms for Hepta data. (a) Input data. (b) Conventional SOM. (c) Growing Grid. (d) FN-SOM.



(a) (c)

Fig. 4. U-Matrix of simulation results for Iris data. Label 1, 2 and 3 correspond to *Iris setosa*, *Iris versicolor* and *Iris virginica*, respectively. (a) Conventional SOM. (b) FN-SOM.

TABLE III

QUANTIZATION ERROR Qe, TOPOGRAPHIC ERROR Te AND NEURON UTILIZATION U FOR IRIS DATA.

	SOM	Growing Grid	FN-SOM
Qe	0.0374	0.0452	0.0323
Te	0.2400	0.2267	0.1667
U	0.7200	0.7048	0.8200

the sepal width, the petal length and the petal width, namely, the input data are 4-dimension. The three classes correspond to *Iris setosa*, *Iris versicolor* and *Iris virginica*, respectively. *Iris setosa* is linearly separable from the other two, however *Iris versicolor* and *Iris virginica* are not linearly separable from each other.

We repeat the learning 100 times for all input data, namely $t_{\max} = 15000$. The input data are normalized and are sorted at random. The learning conditions are the same used in Fig. 2.

Figure 4 shows distances between neighboring neurons and thus visualizes the cluster structure of the map [5]. We can see that the boundary line of FN-SOM is clearer than SOM.

The calculated quantization error Qe, the topographic error Te and the neuron utilization U are shown in Table. III. We confirm that Qe and Te of FN-SOM are the smallest value in the three algorithms. Qe of FN-SOM has improved 13.6% from using the conventional SOM and Te of FN-SOM has also improved 30.5% from using the conventional SOM. This is because the result of FN-SOM hardly has inactive neurons between *Iris setosa* and the other two, therefore, the more neurons can self-organize the data of *Iris versicolor* and *Iris virginica*. Furthermore, U of FN-SOM is also the best value. From these results, we can confirm the efficiency of FN-SOM.

IV. CONCLUSIONS

In this study, we have proposed a new SOM algorithm, SOM with False-Neighbor degree between neurons (called FN-SOM). False-neighbor degrees are allocated between adjacent rows and adjacent columns of FN-SOM. The initial values of all of the false-neighbor degrees are set to zero, however, they are increased with learning, and the false-neighbor degrees act as a burden of the distance between map nodes when the weight vectors of neurons are updated. We have applied FN-SOM to 2-dimensional data, 3-dimensional data and Iris data, and we have investigated the learning behaviors of FN-SOM. Furthermore, the results were compared with those obtained by the conventional SOM and Growing Grid. We have confirmed that the quantization error and the topographic error of FN-SOM were the smallest value in the three algorithms. Moreover, the neuron utilization of FN-SOM was the largest value in the three algorithms. From these results, we have confirmed the efficiency of FN-SOM.

ACKNOWLEDGMENT

This work was partly supported by Yamaha Music Foundation.

REFERENCES

[1] T. Kohonen, *Self-organizing Maps*, Berlin, Springer, 1995.
[2] B. Fritzke, “Growing Grid – a self-organizing network with constant neighborhood range and adaptation strength,” *Neural Processing Letters*, vol. 2, no. 5, pp. 9–13, 1995.
[3] Y. Cheung and L. Law, “Rival-Model Penalized Self-Organizing Map,” *IEEE Trans. Neural Networks*, vol. 18, no. 1, pp. 289–295, 2007.
[4] H. Matsushita and Y. Nishio, “Self-Organizing Map Considering False Neighboring Neuron,” *Proc. of ISCAS’07*, pp. 1533–1536, 2007.
[5] A. Ultsch, “Clustering with SOM: U*C,” *Proc. of WSOM’05*, pp. 75–82, 2005.
[6] K. Kiviluoto, “Topology Preservation in Self-Organizing Maps,” *Proc. of ICNN*, pp. 294–299, 1996.
[7] R. A. Fisher, “The Use of Multiple Measurements in Taxonomic Problems,” *Annual Eugenics*, no.7, part II, pp. 179–188, 1936.